UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/691,759	10/23/2003	Idan Avraham	MSFT-2816/305955.01	6719

41505 7590 11/09/2009

WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION)
CIRA CENTRE, 12TH FLOOR
2929 ARCH STREET
PHILADELPHIA, PA 19104-2891

EXAMINER	
SHIH, HAOSHIAN	

ART UNIT	PAPER NUMBER
2173	

MAIL DATE	DELIVERY MODE
11/09/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/691,759	AVRAHAM ET AL.	
	Examiner	Art Unit	
	HAOSHIAN SHIH	2173	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 August 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-30 are pending in this application and have been examined in response to application RCE filed on 08/13/2009.

2. The previously applied rejection under USC 112 is hereby withdrawn in view of applicant's amendment.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 26-28 are rejected under 35 U.S.C. 101 because a "computer-readable medium" is being recited; the recited "computer-readable medium" may be an electromagnetic signal (Application spec. [0034]). This subject matter is not limited to storage, it also propagate or transmits signals. Thus, non-statutory.

5. Claims 29-30 are rejected under 35 U.S.C. 101 because a "system" is being recited; the recited "system" may be a computer software that performs various functions (Application spec. [0037], "operating system"). Thus, the recited "system" is software per se and not a process, a machine, a manufacture or a composition of matter. Accordingly, the claim fails to recite statutory subject mater as defined in 35. USC 101.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claim 13 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

8. Claim 13 recites the limitation "a host window manager" in line 2. There is insufficient antecedent basis for this limitation in the claim. The Examiner suggests "the host window manager".

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claim 1, 3-4, 12-13, 15, 17-18 and 26-27 are rejected under 35 U.S.C. 102(b) as being anticipated by Boebert et al. (Boebert, US 5,822,435).

11. As to **independent** claim 1, Boebert discloses a method for maintaining the security of data displayed on a display for a system comprising a secured execution environment and a second execution environment the method comprising: operating, on the system the second execution environment concurrently with the secured execution environment, the secure execution environment comprising a nexus and the second

execution environment comprising a different operating system (fig.2; col.4, lines 33-45; a multilevel trusted path operating system "30", "60" is running with a second/workstation operating system "40" to form a secure networked computer system; fig.6; col.9, lines 7-16; during a secured/trusted execution environment, secured/trusted information overlays some parts of the second/workstation environment, displaying both information concurrently on the display), wherein the nexus and different operating system share a window manager (fig.5; col.8, lines 50-63; both the secure execution environment and the second execution environment shares a video manager "34", wherein a video multiplexer "76" within in the video manager "34" is used during the second execution environment, and a video RAM "74" within in the video manager "34" is used during the secure execution environment);

storing an image of at least one graphical user interface element of said nexus, said at least one nexus graphical user interface element (col.5, lines 33-36; "trusted window") being associated with a first process running on said secured execution environment (col.5, lines 14-18; "trusted path mode"); and

displaying said nexus graphical user interface element on said display completely on a display, such that no part of said nexus graphical user interface element is obscured by a graphical user interface element associated with said different operating system (fig.2, a trusted subsystem "67" that includes a cryptographic entity "69" is different from a untrusted subsystem "63"; col.4, lines 51-55) of said second execution environment on said display (col.5, lines 33-43; no parts of the nexus GUI is obscured because the nexus GUI is "overlaid" on top of the screen display.).

12. As to claim 3, Boebert discloses displaying said nexus graphical user interface element such that no part of said nexus graphical user interface element is obscured by a graphical user interface element associated with a second process running on said secured execution environment (col.5, lines 33-43; no parts of the nexus GUI is obscured because the nexus GUI is "overlaid" on top of the screen display).

13. As to claim 4, Boebert discloses displaying only graphical user interface elements on display upon receipt of a user secure display indication (col.5, lines 27-32).

14. As to **independent** claim 12, Boebert discloses a method for maintaining the security of data displayed on a display for a system comprising a secured execution environment and a second execution environment, the method comprising: operating, on the system, the second execution environment concurrently with the secured execution environment, the secure execution environment comprising a nexus and the second execution environment comprising a different operating system, wherein the nexus and different operating system share a window manager (fig.2, a trusted sub operating system "67" that includes a cryptographic entity "69" is different from an untrusted sub operating system "63"; col.4, lines 51-55; col.5, lines 34-42; col.8, lines 45-50; during a secured/trusted execution environment, secured/trusted information

overlays some parts of the unsecure/untrusted environment, displaying both information concurrently);

storing public title information and private title information for a graphical user interface element of said nexus, the nexus graphical user interface element being associated with a process running on said secured execution environment; using said private title information for window management functions on said secured execution environment when displaying said nexus graphical user interface element; and providing said public title information for use in said second execution environment (col.5, lines 33-43; col.7, lines 20-25; col.8, lines 45-50; private title information is contained in secret information, and the public title information is contained in the unclassified information in order to prevent data of different security level from being mixed).

15. As to claim 13, Boebert discloses the window manager comprising a host window manager (fig.5; col.8, lines 50-63; "video manager"), where said second execution environment includes the host window manager for managing graphical user interface elements on said display, where said host window manager creates a shadow graphical user interface element for said nexus graphical user interface element, and where said public title is used by said host window manager (col.5, lines 33-43; col.7, lines 20-25; col.8, lines 45-50; private title information is contained in secret information, and the public title information is contained in the unclassified information in order to prevent data of different security level from being mixed.).

16. As to **independent** claim 15, see rationale addressed in the rejection of claim 1 above.

17. As to claim 17, see rationale addressed in the rejection of claim 3 above.

18. As to claim 18, see rationale addressed in the rejection of claim 4 above.

19. As to **independent** claim 26, see rationale addressed in the rejection of claim 1 above.

20. As to claim 27, see rationale addressed in the rejection of claim 13 above.

Claim Rejections - 35 USC § 103

21. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

22. Claims 2 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boebert and of Janssen et al. (Janssen, US 6,512,529 B1).

23. As to claim 2, Boebert does not disclose ensuring that nexus graphical user interface element contains no areas of transparency.

In the same field of endeavor, Janssen discloses a graphical user interface element contains no areas of transparency. (col.3, lines 23-25);

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Janssen* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include opaque user interface mode taught by Janssen with the motivation being to ensure proper visibility of the secured execution environment.

24. As to claim 16, see rationale addressed in the rejection of claim 2 above.

25. Claims 5-6, 7-8, 10-11, 14, 19-20, 21-22, 24-25 and 28-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boebert and Ye et al. (Ye, "Trusted paths for browsers: An open-source solution to web spoofing", Feb 4, 2002).

26. As to **independent** claim 5, Boebert discloses a method for maintaining the security of data displayed on a display for a system comprising a secured execution environment and a second execution environment the method comprising: operating, on the system, the second execution environment concurrently with the secured execution

environment, the secure execution environment comprising a nexus and the second execution environment comprising a different operating system (fig.2; col.4, lines 33-45; a multilevel trusted path operating system "30", "60" is running with a second/workstation operating system "40" to form a secure networked computer system; fig.6; col.9, lines 7-16; during a secured/trusted execution environment, secured/trusted information overlays some parts of the second/workstation environment, displaying both information concurrently on the display), where the nexus and different operating system share a window manager (fig.5; col.8, lines 50-63; "video manager");

displaying a graphical user interface element , where said nexus graphical user interface element is associated with a process running on said secured execution environment (col.4, lines 4-15). Boebert does not specifically disclose storing and display a nexus-user secret associated with said secured execution environment.

In the same field of endeavor, ye, discloses storing and display a nexus-user secret associated with said secured execution environment (Section 4.2 "Synchronized random dynamic boundaries").

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Ye* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include synchronized random dynamic boundaries taught by Ye with the motivation being to provide an effective trust

judgment about the identity of a graphic interface element in a human-computer interaction environment.

27. As to claim 6, Ye discloses accepting a user nexus-user secret display indication; and displaying said nexus-user secret (Section 4.2 "Synchronized random dynamic boundaries"; the nexus-user secret disclosed here is having trusted and untrusted color borders representing each of the nexus and the second execution environments).

28. As to **independent** claim 7, Boebert discloses a method for maintaining the security of data displayed on a display for a system comprising a secured execution environment and a second execution environment the method comprising: operating, on the system, the second execution environment concurrently with the secured execution environment, the secure execution environment comprising a nexus and the second execution environment comprising a different operating system (fig.2; col.4, lines 33-45; a multilevel trusted path operating system "30", "60" is running with a second/workstation operating system "40" to form a secure networked computer system; fig.6; col.9, lines 7-16; during a secured/trusted execution environment, secured/trusted information overlays some parts of the second/workstation environment, displaying both information concurrently on the display), wherein the nexus and different operating system share a window manager (fig.5; col.8, lines 50-63; "video manager");

accepting at least two graphical data elements of said nexus, each associated with a process running on said secured execution environment, for display on said display; and displaying at least two graphical user interface elements of said nexus, each of said nexus graphical user interface elements comprising one of said nexus graphical data elements (col.6, lines 52- 56). Boebert does not disclose a common graphical user interface decoration.

In the same field of endeavor, Ye discloses a common graphical user interface decoration (Section 4.2 "Synchronized random dynamic boundaries"; same window borders and styles for trusted environment).

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Ye* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include synchronized random dynamic boundaries taught by Ye with the motivation being to provide an effective trust judgment about the identity of a graphic interface element in a human-computer interaction environment.

29. As to claim 8, Ye discloses common graphical user interface decoration comprises a colored border (Section 4.2, "Synchronized random dynamic boundaries"; Section 5.1 "Adding colored boundaries").

30. As to claim 10, Ye discloses changing said common graphical user interface decoration when a set time period elapses (Section 5.2 "Making the boundaries dynamic"; the "setInterval" sets the time interval for a change in the graphical user interface decoration).

31. As to claim 11, Ye discloses changing said common graphical user interface decoration when a user decoration change indication is received (Section 5.2 "Making the boundaries dynamic"; the "example-changeBorder.js" script that is in charge of the border style is set by a user).

32. As to claim 14, Boebert discloses displaying each of said nexus graphical user interface element on said display completely on a display, such that no part of said nexus graphical user interface element is obscured by a graphical user interface element associated with said second execution environment on said display (col.5, lines 33-43; no parts of the nexus GUI is obscured because the nexus GUI is "overlaid" on top of the screen display). Boebert does not disclose each of said nexus graphical user interface elements comprises a common graphical user interface decoration. Storing a nexus-user secret associated with said secured execution environment; and displaying a nexus-user secret graphical user interface element comprising said nexus-user secret on said display.

In the same field of endeavor, Ye discloses each of said nexus graphical user interface elements comprises a common graphical user interface decoration. Storing a nexus-user secret associated with said secured execution environment; and displaying a nexus-user secret graphical user interface element comprising said nexus-user secret on said display (Section 4.2 "Synchronized random dynamic boundaries"; same window borders and styles for trusted environment; the nexus-user secret disclosed here is having trusted and untrusted color borders representing each of the nexus and the second execution environments).

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Ye* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include synchronized random dynamic boundaries taught by Ye with the motivation being to provide an effective trust judgment about the identity of a graphic interface element in a human-computer interaction environment.

33. As to **independent** claim 19, see rationale addressed in the rejection of claim 5 above.

34. As to **independent** claim 21, see rationale addressed in the rejection of claim 7 above.

35. As to claim 20, see rationale addressed in the rejection of claim 6 above.

36. As to claim 22, see rationale addressed in the rejection of claim 8 above.

37. As to claim 24, see rationale addressed in the rejection of claim 10 above.

38. As to claim 25, see rationale addressed in the rejection of claim 11 above.

39. As to claim 28, see rationale addressed in the rejection of claim 14 above.

40. As to **independent** claim 29, Boebert discloses a system for maintaining the security of data displayed on a display, the system comprising: operating the second execution environment concurrently with the secured execution environment, the secure execution environment comprising a nexus and the second execution environment comprising a different operating system (fig.2; col.4, lines 33-45; a multilevel trusted path operating system "30", "60" is running with a second/workstation operating system "40" to form a secure networked computer system; fig.6; col.9, lines 7-16; during a secured/trusted execution environment, secured/trusted information overlays some parts of the second/workstation environment, displaying both information concurrently on the display);

first storage in said secured execution environment for storing private title information for a graphical user interface element of said nexus, the nexus graphical

user interface element being associated with a process running on said secured execution environment and a nexus-user secret associated with said secured execution environment; second storage in said second execution environment for storing public title information for said nexus graphical user interface element; a trusted window manager for displaying said nexus graphical user interface element on said display (col.5, lines 33-43; col.7, lines 20-25; col.8, lines 45-50; private title information is contained in secret information, and the public title information is contained in the unclassified information in order to prevent data of different security level from being mixed), such that no part of said nexus graphical user interface element is obscured by a graphical user interface element associated with said second execution environment on said display(col.5, lines 33-43; no parts of the nexus GUI is obscured because the nexus GUI is "overlaid" on top of the screen display, wherein the nexus and different operating system share the trusted window manager (fig.5; col.8, lines 50-63; "video manager");). Boebert does not disclose where said nexus graphical user interface element comprises a common graphical user interface decoration and said private title information.

In the same field of endeavor, Ye discloses nexus graphical user interface elements comprises a common graphical user interface decoration (Section 4.2 "Synchronized random dynamic boundaries"; same window borders and styles for trusted environment), and private title information (Section 4.2 "Synchronized random dynamic boundaries", secret information such as the border colors, styles and intervals of the

random changes are considered as private title because the private title is used only under a secured execution environment).

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Ye* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include synchronized random dynamic boundaries taught by Ye with the motivation being to provide an effective trust judgment about the identity of a graphic interface element in a human-computer interaction environment.

41. As to claim 30, Ye discloses displaying a nexus-user secret graphical user interface element comprising said nexus-user secret on said display (Section 4.2 "Synchronized random dynamic boundaries"; the nexus-user secret disclosed here is having trusted and untrusted color borders representing each of the nexus and the second execution environments).

42. **Claims 9 and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Boebert , Ye and Dhamija (Dhamija, "Hash visualization in user authentication", April 2000)**

43. As to claim 9, Boebert does not disclose common graphical user interface decoration comprises one or more randomly selected images.

In the same field of endeavor, Ye discloses a common graphical user interface decoration (Section 4.2 "Synchronized random dynamic boundaries"; same window borders and styles for trusted environment).

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and *Ye* before him at the time the invention was made, to modify the secured execution environment interface taught by Boebert to include synchronized random dynamic boundaries taught by Ye with the motivation being to provide an effective trust judgment about the identity of a graphic interface element in a human-computer interaction environment.

Ye does not disclose using one or more randomly selected images.

In the same field of endeavor, Dhamija discloses randomly selected images (Paragraph "A prototype image authentication system");

It would have been obvious to one of ordinary skill in the art, having the teaching of Boebert and Ye, and the teaching of *Dhamija* before him at the time the invention was made, to modify the secured execution environment reorganization interface taught by Boebert and Ye to include random selected images taught by Dhamija with the motivation being to provide an easy to remember and hard to write down trust judgment

about the identity of a graphic interface element in a human-computer interaction environment.

44. As to claim 23, see rationale addressed in the rejection of claim 9 above.

Response to Arguments

45. Applicant's arguments filed 08/13/2009 have been fully considered but they are not persuasive.

46. Applicant argues that Boebert does not disclose a second execution environment operating concurrently on the system.

In response to applicant's argument, Boebert discloses a secure networked computer system (col.4, lines 32) which includes a workstation "40" running a first level execution environment and a multi-level computer "60" with a trusted path subsystem "30" running at least a second level execution environment, a user can invoke the second level execution environment while in the first level execution environment, when the second level execution environment is invoked, the first level execution environment is isolated from the second level execution environment. The first level execution environment and the second execution environment are **running concurrently** on the networked computer system in the sense that during the isolation of the first level execution environment, the first level execution environment is still running even though the first

level execution environment is isolated from the second level execution environment (fig. 2; col.6, lines 60- col.7, lines 11).

47. Applicant argues that Boebert does not disclose the two operating systems share a window manager.

In response to applicant's argument, Boebert discloses having both a secure/trusted execution environment and a second/workstation execution environment sharing a video manager "34", wherein a video multiplexer "76" within in the video manager "34" is used during the second/workstation execution environment, and a video RAM "74" within in the video manager "34" is used during the secure/trusted execution environment (fig.5; col.8, lines 50-63).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HAOSHIAN SHIH whose telephone number is (571)270-1257. The examiner can normally be reached on m-f 0730-1700.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kieu Vu can be reached on (571) 272-4057. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

HSS

/Kieu Vu/
Supervisory Patent Examiner, Art Unit 2173